Final Report

The Design of Self-Checking Reduced Instruction Set Computers

ONR Grant #: N00014-91-J-1067

Dr. T.R.N. Rao

Center for Advanced Computer Studies
University of Southwestern Louisiana
Lafayette, LA

Submitted to:
Dr. Clifford Lau
Department of the Navy
Office of Naval Research
Arlington, VA

Report Period: October 1, 1990 thru March 31, 1993

Final Report

The overall goal of this project is the design of self-checking computers and fault-tolerant memory systems with low overhead and high reliability. During the research, we focused our research on the determination of more efficient approach for realization of self-checking systems and developing new approach for fault-tolerant memory systems. The research conducted in this project is summarized under five different topics given below. Fifteen papers have been published by the PI under the contract and were listed in Section 6. Abstracts of these papers are included at the end of this report.

1. Self-Checking System Designs

1.1 Totally Self-Checking Systems

We investigated designs of Self-Checking systems. This lead us to a new viewpoint that a totally self-checking (TSC) system can be regarded as a connection of TSC subsystems without using checkers at the embedded interfaces except at the primary output of the system. That is, each subsystem is Totally Self-Checking and Code-Disjoint (TSC-CD). By this motivation, we studied the problem of designing TSC-CD Programmable Logic Array (PLA) for combinational circuits. We have introduced a design method of TSC-CD PLA for combinational circuits. This method generalized the realization of CD combinational circuits, which was proposed by Nanya et al. in 1989. The hardware overhead will be reduced significantly using this method. For example, the input code and the output code are the Berger codes with length ($n + \lfloor \log_2 n \rfloor + 1$), the additional hardware requirement is about $\frac{2\cdot(\lfloor \log_2 n \rfloor + 1)}{n}$. In addition, we also derived an algorithm for choosing $D(I_i,C_i)$ function (for detail see [6]). This algorithm is very similar to the Quine-MaCluskey algorithm for formal minimization procedure for two-level Boolean expressions. Once the vectors $D(I_i,C_i)$ are

chosen, using an algorithm to simplify a multiple output Boolean function into a near minimal sum-of-products for PLAs, the TSC-CD PLA of combinational circuit is designed.

This design method can also be applied to that of TSC-CD Moore type sequential circuits, in which the next-state functions and the output functions will be implemented by the PLA. On the other hand, TSC checkers are all TSC-CD combinational circuits. Therefore, the design method has also solved the problem of designing the checkers with PLA.

The results of TSC-CD PLA design for combinational circuits has been presented as a paper at the *Indian Computing Congress 1991*.

1.2 Strongly Fault-Secure Systems

However, the combination of a TSC and CD circuit is more difficult to implement than strongly fault-secure (SFS) and strongly code-disjoint (SCD) circuit. Generalizing the previous theory to the SFS-SCD combinational circuits, we have developed a PLA design scheme to achieve both SFS and SCD. Each SFS and SCD combinational circuit is formed by two PLAs. The first PLA receives only the information part of input to generate the information part of output and $k - r$ least signification check bits of the output, where k is the check bit length. The second PLA receives both information part and check part of input to generate the r most significant check bits of the output for $r \geq 1$. The selection of r depends on the implemented function to achieve a higher implementation efficiency. We also compared this design with a recent design by Nanya and Uchida and found that this design was simple in structure and faster in error checking. This design details can be found in the a technical report " *The Design of Strongly Fault-Secure and Strongly Code-Disjoint PLAs* ".

2. Berger Check Prediction and Berger Checker

2.1 Berger Check Prediction and Reduced Berger Check Prediction

We have developed a theory for Berger check prediction (BCP). By this theory the Berger code is applied to the arithmetic and logic unit (ALU), then the entire processor is Berger encoded and this would result in the most efficient design. A novel scheme to design concurrent-error-detecting ALU and a strongly fault-secure ALU design based on the BCP technique have been written in two papers, which have been published in *IEEE Trans. on Computers* and *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, respectively. Recently, we further developed a theory for reduced Berger check prediction (RBCP). By this new theory, a reduced Berger code, which uses only the two least significant check bits of its Berger code, is used to encode both operands and the computation result for ALU. Since a Berger code requires $\lceil \log_2(n+1) \rceil$ check bits for n information bits, the application of reduced Berger code yields more efficient implementation of a strongly fault-secure ALU then the previous BCP. A paper was written on these results and has been published in 1993 Proceeding of *Fifth Annual IEEE International Conference on Wafer Scale Integration* pp.163-172.

2.2 A New Design of Berger Code Checker

Berger codes are the only known systematic all unidirectional error detecting codes and they are widely applied in fault-tolerant systems. Therefore, designing a better Berger code checker is very important. The most well known structure of Berger code checker is usually referred to as normal checker and is to generate the replicated check bits of value complementary to the original ones and to compare them by a two-rail comparator. In 1989, a design of totally self-checking Berger code checker using a Berger code partitioning scheme has been proposed by Piestrak. This design proceeds from the idea that any Berger code can be constructed from $u = \lceil (|I| + 1)/2 \rceil$ *m*-out-of-*n*

codes, where $|I|$ is the number of information bits, $m = 1, 3, 5, \ldots, 2u-1$ and $n = |I| + 1$. Compared with the normal checker, this method offers an improvement in delay but with a great cost in hardware.

We generalized Berger code partitioning scheme and applied this method to the Programmable logic array based design of a TSC Berger code checker. The Piestrak's Berger code partitioning scheme is a special case of our generalized Berger code partitioning scheme. We also proved that any Berger code can be converted, if necessary, to a suitable 2^s form, i.e., any Berger code can be constructed from m-out-of-2^{s+1} codes. Furthermore, we also developed a design of m-out-of-2^{s+1} code checkers via PLA, which have a regular structure and a reduced circuit complexity $O(n^2)$ for codewords of length n.

These results have been written in an extended abstract of a paper " The design of Totally Self-Checking Berger Code Checkers Based on Generalized Berger Code Partition ", which was presented at *ONR Dependable Computing Workshop* Nov. 12-13, 1991, and this paper has been accepted for publication in *IEEE Trans. on Computers.*

3. Fault-Tolerant Memory Systems

For a Self-Checking computer or processor, fault-tolerant memory system is a important part. In some computer memory systems in which the data are stored in a byte-per-chip or byte-per-card fashion, the errors are likely to be confined to one or a few bytes. For this kind of computer memory system, it is often probable that when errors occur in multiple bytes the errors will be unidirectional within each individual byte. However, the errors in one byte may be of the form $1 \rightarrow 0$ while in another byte they may be of the form $0 \rightarrow 1$. When a memory word is handled as a whole, unidirectional errors of a single form may occur across the entire word.

The codes, which can detect some small number t of unidirectional byte errors in computer memory words composed of small information bytes each containing b bits, and simultaneously detect all unidirectional errors across the entire memory word, were developed by Dunning, Dial and Varanasi. These codes always use two additional b-bit bytes to hold parity check information and were said to be t-unidirectional byte error detecting and all unidirectional error detecting (t-UbED/AUED) codes. We generalized the construction of the t-UbED/AUED codes proposed by Dunning et al. to any t and improved these codes in the case of $t \geq 3$. Thus, more efficient t-UbED/AUED codes are found. These results have been written as a paper "Efficient Multiple Byte Unidirectional Error-Detecting Codes for Computer Memory Systems", which was published in Proceeding of *The Twenty Second International Symposium on Fault-Tolerance Computing.*

4 Algorithm-Based Fault Tolerance Systems

Algorithm-based fault tolerance (ABFT) is a scheme which improves the reliability of parallel systems with very low overhead compared to other fault tolerance schemes with similar benefits. It was proposed by Huang and Abraham for parallel matrix operations in 1984. This scheme is distinguished by three characteristics: (1) encode data at a higher level; (2) redesign algorithms to operate on the encoded data; (3) distribute the computation steps of the redesigned algorithm among computation processors, such that failure of any processor affects as few data as possible. Possessing these characteristics, ABFT is a novel system-level method of achieving high reliability. Since 1984, a lot of papers applying this scheme to parallel systems have appeared.

In order to efficiently apply ABFT to more parallel systems, the encoding of data such that multiple errors, especially a group of errors, can be detected or corrected is a central problem. Unfortunately, few papers have addressed this problem. We investigated the problem of encoding of data for ABFT systems and have developed a new encoding scheme. The significance of this scheme is

that any linear binary error-correcting code can be used to derive an error-correcting code over Z_{2^m} with the same length and rate. Since in coding theory, lot of work about choosing better binary linear codes and decodings of the better codes has been done, using this scheme the efficient linear error-correcting codes over Z_{2^m} with any parameters can be easily found. We have also introduced two kinds of error-correcting codes over Z_{2^m}. They are the BCH-like codes and the Reed-Solomon-like codes, which are derived by the BCH codes and the Reed-Solomon codes, respectively. These two kind of codes over Z_{2^m} are very useful to the ABFT systems. We also developed two decoding procedures for these two kinds of codes. The results have been written as a paper "Error Correcting Codes over Z_{2^m} for Algorithm-Based Fault Tolerance, which has been accepted for publication in *IEEE Trans. on Computers* .

5. Coding Theory

5.1. Decoding for Algebraic Geometric Codes

The introduction of algebraic geometric codes is the most important development in the theory of error-correcting codes in the past ten years. Tsfasman, Vladut and Zink (1982) showed an extremely exciting result, that is, the existence of a sequence of codes, which exceeds the Gilbert-Varshamov bound. For this paper they received the IEEE Information Theory Group Paper Award for 1983. Since then, many papers dealing with algebraic geometric codes and their decoding procedures have appeared. Good code constructions are very important. Moreover, it is desirable and important to derive simple decoding procedures which can correct as many errors as possible. However, any simple decoding procedure, which can correct errors up to $\lfloor \frac{d^*-1}{2} \rfloor$, has not been presented yet, where d^* is the designed minimum distance of the algebraic geometric code.

We investigated this problem and derived a simple decoding procedure for algebraic geometric codes, which can correct any $\lfloor \frac{d^*-1}{2} \rfloor$ or less errors with a complexity $O(n^3)$. This decoding procedure employs a modified fundamental iterative algorithm that has been introduced to derive the Berlekamp-Massey algorithm, and utilizes the error-correcting capability of the code. In principle, it is a generalization of Peterson's decoding procedure for the BCH codes. This decoding procedure, called as Feng-Rao decoding procedure, is considered as a major breakthrough in this field and has been written as a paper, which was presented at the *Ninth International Symposium on Applied Algebra, Algebraic Algorithms and Error-Correcting Codes* and published in *IEEE Trans. on Information Theory* pp. 37-45, Jan. 1993. We also investigated a fast decoding procedure for algebraic-geometric codes, which can correct any $\lfloor \frac{d^*-1}{2} \rfloor$ or less errors with a complexity $O(n^{7/3})$. This decoding procedure has been written as a paper and this paper has been accepted for publication in *IEEE Trans. on Information Theory*. We also have developed an erasures-and-errors decoding procedure for algebraic-geometric codes. The result has been published at *the 1993 IEEE Information Theory Workshop* as an invited talk.

5.2. A New Approach for Algebraic Geometric Codes

The current algebraic-geometric codes are based on the theory of algebraic geometric curves. We investigated a simple approach for the construction of algebraic-geometric codes which does not require an extensive background in algebraic geometry. Given an affine plane irreducible curve and all its rational points, we can find a sequence of monomials $x^i y^j$ based on the equation of the curve. Using the first r monomials as a basis for the dual code of a linear code, the designed minimum distance d of the linear code, called algebraic-geometric code, can be easily determined. For these codes, we show a fast decoding procedure with a complexity $O(n^{7/3})$, which can correct errors up to $\lfloor (d-1)/2 \rfloor$. For this approach it is neither necessary to know the genus of curve nor the basis of a

differential form. This approach can be easily understood by most engineers. In this research, we have obtained a new improvement of the Goppa bound, called the Feng-Rao bound. We also generalized this result to the codes from the curves in high-dimensional speces. These results are written as two papers " A Simple Approach for Construction of Algebraic Geometric Codes from Affine Plane Curves" and "A Class of Algebraic Geometric Codes from Curves in High-Dimensional Spaces", which were presented at *IEEE International Symposium on Information Theory 1993* and *10th International Symposium, Applied Algebrais Algorithms and Error-Correcting Codes*, respectively. The first paper will appear in *IEEE Trans. on Information Theory* and the second paper has been published in *Lecture Notes in Computer Science 673* pp. 132-146. May,1993.

6. Publications of the PI's (1990-1993)

[1] K. Sakaniwa, T. N. Ahn, and T. R. N. Rao, "A Note on t-Unidirectional Error Correcting and d (d≥t)-Unidirectional Error Detecting (t-UEC and d-UED) Codes" *IEEE Trans. on Computers*, Vol. C-40, pp. 987-988, Aug. 1991.

[2] T. L. Hwang and T. R. N. Rao, "Secret Error-Correcting Codes (SECC)" *Journal of the Institute of Electronics and Telecommunication Engineering*, Vol. 36, pp. 362-376, 1990.

[3] K. C. Zeng, C. H. Yang, D. Y. Wei, and T. R. N. Rao, "Pseudorandom Bit Generators in Stream-Cipher Cryptography" *IEEE Computer*, pp. 8-17, Feb. 1991.

[4] K. C. Zeng, D. Y. Wei, and T. R. N. Rao, "d-Functions in $V_k(F_2)$ and Self-Decimation of m-Sequences", *Lecture Notes in Computer Science-539*, pp.465-476, 1991.

[5] J. C. Lo, S. Thanawastien, T. R. N. Rao, and M. Nicolaidis, "An SFS Berger Check Prediction ALU and Its Application to Self-Checking Processor Designs" *IEEE Trans. on CAD*. pp.525-540, April, 1992.

[6] T. R. N. Rao, G. L. Feng, and D. Y. Wei, "A Design Method for Totally Self-Checking and Code-Disjoint PLA of combinational circuits" *Indian Computing Congress 1991.*

[7] G. L. Feng and T. R. N. Rao, "Decoding Algebraic Geometric Codes up to the Design Minimum Distance" *AAECC-9, 1991* and *IEEE Trans. on Infor. Theory.* Vol. IT-39, No.1, pp. 37-45, Jan. 1993.

[8] G. L. Feng and T. R. N. Rao, "Error Correcting Codes over Z_{2^m} for Algorithm-Based Fault Tolerance" Accepted for publication in *IEEE Trans. on Computers.*

[9] T.R.N. Rao, G.L. Feng, M.S. Kolluru, and J.C. Lo, "Novel Totally Self-Checking Berger Code Checker Designs Based on Generalized Berger Code Partitioning," Accepted for publication in *IEEE Trans. on Computers.*

[10] T. R. N. Rao, G. L. Feng, and M. S. Kolluru, "Efficient Multiple Byte Unidirectional Error-Detecting Codes for Computer Memory Systems" *Dig. 22nd Fault Tolerant Computing Symposium, 1992..* pp. 502-509. (1992)

[11] G.L. Feng, V.K. Wei, T.R.N. Rao, and K.K. Tzeng, "True Designed-Distance Decoding of a Class of Algebraic-Geometric Codes, Part I: A New Theory without Riemann-Roch Theorem," Accepted for publication in *IEEE Trans. on Infor. Theory.*

[12] G.L. Feng, V.K. Wei, T.R.N. Rao, and K.K. Tzeng, "True Designed-Distance Decoding of a Class of Algebraic-Geometric Codes, Part II: Fast Algorithm and Block-Hankel Matrices," Accepted for publication in *IEEE Trans. on Infor. Theory.*

[13] J. H. Kim, T. R. N. Rao, G. L. Feng, and J. C. LO, "The Efficient Design of a Strongly Fault-Secure ALU Using a Reduced Berger Code for WSI Processor," *1993 Proceeding of 5th Annual IEEE International Conference on Wafer Scale Integration,* PP. 163-172.

[14] G.L. Feng and T.R.N. Rao, "A Simple Approach for Construction of Algebraic Geometric Codes from Affine Plane Curves," Presented at *the IEEE International Symposiums on Information Theory*, Jan. 19-24, 1993 and to appear in *IEEE Trans. on Infor. Theory*..

[15] G.L. Feng and T.R.N. Rao, "A Class of Algebraic Geometric Codes from Curves in High-Dimensional Projective Spaces," *Lecture Notes in Computer Science*, pp. 132-146, May, 1993.

[16] G.L. Feng and T.R.N. Rao, "Erasures-and-Errors Decoding of Algebraic-Geometric Codes," invited talk at *1993 IEEE Information Theory Workshop* June, 4-8, 1993 in Japan.

Correspondence

A Note on t-Unidirectional Error Correcting and $d(d \geq t$ -Unidirectional Error Detecting (t-UEC and d-UED) Codes

Kohichi Sakaniwa, Tae Nam Ahn, and T. R. N. Rao

Abstract— **This correspondence shows necessary and sufficient conditions for t-unidirectional error correcting and d-unidirectional error detecting (t-UEC and d-UED) codes and corrects an error in a theorem previously published on t-UEC and d-UED codes [1].**

Index Terms— **Asymmetric errors, error correction, error detection, error protection, Hamming distance, unidirectional errors.**

I. Introduction

Unidirectional error correcting and/or detecting codes have been extensively investigated by several authors [1]–[3], etc., and have provided a powerful tool for error protection in LSI memories in which the most likely faults cause unidirectional errors [1]–[3]. In Theorem 2.1 of Lin and Bose [1], the authors have given a necessary and sufficient condition for a binary code C to be a t-unidirectional error correcting and $d(d > t)$-unidirectional error detecting (t-UEC and d-UED) code. In this correspondence, we adopt the same notation as in [1]. In [1] the necessary and sufficient conditions for a binary t-UEC and d-UED code C are given as

$$\left.\begin{array}{ll} \text{a} & D(\boldsymbol{X},\boldsymbol{Y}) = N(\boldsymbol{X},\boldsymbol{Y}) + N(\boldsymbol{Y},\boldsymbol{X}) \geq t+d+1, \text{ or} \\ \text{b} & \min\{N(\boldsymbol{X},\boldsymbol{Y}), N(\boldsymbol{Y},\boldsymbol{X})\} \geq t+1 \end{array}\right\} \tag{1}$$

for all $\boldsymbol{X} = (X_0, X_1, \cdots, X_{n-1})$, $\boldsymbol{Y} = (Y_0, Y_1, \cdots, Y_{n-1})$, $\boldsymbol{X} \neq \boldsymbol{Y}$ belonging to C, where $N(\boldsymbol{X},\boldsymbol{Y})$ denotes the number of $1 \to 0$ crossovers from $\boldsymbol{X}$ to $\boldsymbol{Y}$, i.e., $N(\boldsymbol{X},\boldsymbol{Y}) = \sum_{i=0}^{n-1} X_i \overline{Y}_i$, and $D(\boldsymbol{X},\boldsymbol{Y})$ denotes the Hamming distance between $\boldsymbol{X}$ and $\boldsymbol{Y}$. First, we show by means of a counter-example that condition (1) is only *sufficient* but is not *necessary*. Then we state and prove the necessary sufficient conditions for t-UEC and d-UED codes, and finally give some concluding remarks of how this result complements the previously published research [2], [3].

II. Necessary and Sufficient Condition

We first show by means of a counter-example that condition (1) is only *sufficient* but not *necessary*.

Counter-example: Consider a simple code of length 5, $C = \{\boldsymbol{X} = (11110), \boldsymbol{Y} = (00001)\}$. It is easily observed that $D(\boldsymbol{X},\boldsymbol{Y}) = 5$, $N(\boldsymbol{X},\boldsymbol{Y}) = 4$, and $N(\boldsymbol{Y},\boldsymbol{X}) = 1$. Neither a) nor b) of Condition (1) given above holds for $t = 2$ and $d = 3$. However, it is easy to show that C is a 2-UEC and 3-UED code. Indeed, it is also a 3-UEC code as can be checked by an array decoding table.

Manuscript received September 15, 1988; revised April 10, 1990. The work of T. R. N. Rao is supported in part by the National Science Foundation MIPS-8807761 and the Louisiana Board of Regents under Grant 86-USL(2)-127-03.

K. Sakaniwa is with the Department of Electrical and Electronic Engineering, Tokyo Institute of Technology, O-okayama, Meguro-ku, Tokyo 152, Japan.

T. N. Ahn is with the Computing Center, Korea Army Headquarters, Seoul, Korea.

T. R. N. Rao is with the Center for Advanced Computer Studies, University of Southwestern Louisiana, Lafayette, LA 70504.

IEEE Log Number 9042299.

The necessary and sufficient condition for t-UEC and d ($d \geq t$ -UED code is given next by the following theorem.

Theorem: A code C is a t-UEC and d ($d \geq t$ -UED code *if and only if* for any distinct $\boldsymbol{X}, \boldsymbol{Y} \in C$,

$$\left.\begin{array}{ll} \text{a)} & N(\boldsymbol{X},\boldsymbol{Y}) + N(\boldsymbol{X},\boldsymbol{Y}) \geq t-d+1, \text{ or} \\ \text{b)} & \min\{N(\boldsymbol{X},\boldsymbol{Y}), N(\boldsymbol{Y},\boldsymbol{X})\} \geq t-1, \text{ or} \\ \text{c)} & \max\{N(\boldsymbol{X},\boldsymbol{Y}), N(\boldsymbol{Y},\boldsymbol{X})\} \geq d-1, \text{ and} \\ & \min\{N(\boldsymbol{X},\boldsymbol{Y}), N(\boldsymbol{Y},\boldsymbol{X})\} \geq 1. \end{array}\right\} \tag{2}$$

Proof: The sufficiency of Condition (2) is fairly obvious, since for any distinct $\boldsymbol{X}, \boldsymbol{Y} \in C$, we have the following.

1) Condition a) is necessary and sufficient for t-EC and d-ED codes (Theorem 3 of [2]). Therefore, a) is also sufficient for t-UEC and d-UED codes.

2) Condition b) is necessary and sufficient for t-EC and all-UED codes (Theorem 7 of [2]). Therefore, b) is also sufficient for t-UEC and d-UED codes.

3) Condition c) is the second half of necessary and sufficient conditions for d-UEC codes (Theorem 6 of [2]). Therefore, c) is also sufficient for t-UEC and d-UED codes.

To prove the necessity of conditions (2), we give an indirect proof. That is, we assume that condition (2) does not hold for C and prove that C cannot be t-UEC and d-UED. By negating condition (2), we get that for some distinct $\boldsymbol{X}, \boldsymbol{Y} \in C$,

$$\left.\begin{array}{l} N(\boldsymbol{X},\boldsymbol{Y}) + N(\boldsymbol{Y},\boldsymbol{X}) \leq t+d \quad \text{and} \\ \min\{N(\boldsymbol{X},\boldsymbol{Y}), N(\boldsymbol{Y},\boldsymbol{X})\} \leq t \quad \text{and} \\ \max\{N(\boldsymbol{X},\boldsymbol{Y}), N(\boldsymbol{Y},\boldsymbol{X})\} \leq d \end{array}\right\} \tag{3}$$

or

$$\left.\begin{array}{l} N(\boldsymbol{X},\boldsymbol{Y}) + N(\boldsymbol{Y},\boldsymbol{X}) \leq t+d \quad \text{and} \\ \min\{N(\boldsymbol{X},\boldsymbol{Y}), N(\boldsymbol{Y},\boldsymbol{X})\} \leq t \quad \text{and} \\ \min\{N(\boldsymbol{X},\boldsymbol{Y}), N(\boldsymbol{Y},\boldsymbol{X})\} = 0. \end{array}\right\} \tag{4}$$

Note that in condition (4), the middle part is clearly redundant and that the last part implies that $\boldsymbol{X}$ and $\boldsymbol{Y}$ are *ordered*.

First, we assume that condition (3) holds in C. We need to note that if a code C is not t-asymmetric error correcting and d-asymmetric error detecting (t-AEC and d-AED), then it cannot be t-UEC and d-UED either.

We assume without any loss of generality that $N(\boldsymbol{X},\boldsymbol{Y}) \leq N(\boldsymbol{Y},\boldsymbol{X})$. Then condition (3) implies that $N(\boldsymbol{X},\boldsymbol{Y}) = t^* \leq t$, $N(\boldsymbol{Y},\boldsymbol{X}) = d^* \leq d$ ($t^* \leq d^*$) and for $\boldsymbol{X} = (X_0, X_1, \cdots, X_{n-1})$, $\boldsymbol{Y} = (Y_0, Y_1, \cdots, Y_{n-1})$, we have

$$X_i = 1 \quad \text{and} \quad Y_i = 0, \text{ for } i \in I \triangleq \{i_1, i_2, \cdots, i_{t^*}\} \subseteq \{0, 1, 2, \cdots, n-1\}$$

$$X_j = 0 \quad \text{and} \quad Y_j = 1, \text{ for } j \in J \triangleq \{j_1, j_2, \cdots, j_{d^*}\} \subseteq \{0, 1, 2, \cdots, n-1\}$$

where $I \cap J = \emptyset$. Obviously, the remaining positions of $\boldsymbol{X}$ and $\boldsymbol{Y}$ are equal. That is,

$$X_k = Y_k \qquad \text{for } k \notin I \cup J.$$

Consider error vectors $\boldsymbol{E}_{t^*}$ and $\boldsymbol{E}_{d^*}$ as follows:

$$\boldsymbol{E}_{t^*} = (E_0, E_1, \cdots, E_{n-1}),$$

$$\text{where } E_i \triangleq \begin{cases} e_i (1 - \text{error}), & \text{for } i \in I \\ 0 (\text{no error}), & \text{otherwise} \end{cases}$$

0018-9340/91/0800-0987$01.00 © 1991 IEEE

This work was supported in part by the Office of Naval Research under Grant N00014-91-J-1067.

Reprinted form the Journal of the Institution of Electronics and Telecommunication Engineers, Vol. 36, Nos. 5 & 6, 1990, pp. 362-376

Secret Error-Correcting Codes (SECC)

TZONELIH HWANG
National Cheng-Kung University, Institute of Information Engineering, Tainan, Taiwan, ROC

AND

T R N RAO
University of Southwestern Louisiana, The Center of Advanced Computer Studies, Lafayette, Louisiana 70504, USA

As computer systems are expanding to many applications, the assurance of reliable and secure data communications has become an important issue. The conventional approach to achieving this purpose is very inefficient. Secret error-correcting codes (SECC) are designed to solve this problem in one enciphering process. In the SECC, only the authorized user can correct channel errors systematically. Therefore, the presence of channel errors would only increase the security of the system. A block SECC encryption scheme using *nonlinear* codes is proposed to realize this new concept. The SECC scheme given here can also be used to augment an already enciphered text, such as DES ciphertext, to obtain a stronger cipher as well as correction of channel errors.

Indexing terms : Algebraic-code cryptosystem, Joint encryption and error correction, Secret error-correcting codes

THE demand for reliable, secure and efficient digital data transmissions and storage systems has been accelerated by the emergence of large-scale and high speed communication networks. In 1948, Shannon demonstrated that errors induced by a noisy channel or proper encoding of the information [1]. Since Shannon's work, a great deal of developments have contributed toward achieving data reliability and the use of coding for error control has become an integral part of modern communication systems and digital computers.

Information stored in computer systems is particularly vulernable to eavesdropping. Although information can be protected by several ways (*eg*, physical control data are stored in physically secure place; or computer system control the operating system provides access control mechanisms to check user's authentication), data encryption is the only cost-effective way to provide data secrecy [2-7].

As computer systems are expanding to many applications, the assurance of both data reliability and data secrecy has become an important issue. To achieve this purpose, conventionally the first step is to encipher a plaintext (M) into a ciphertext and the second step is to encode the ciphertext into a codeword (C) using an algebraic code. To recover the plaintext (M), the receiver decodes the received word (C' = C + Noise) first and then deciphers the ciphertext (Fig 1). Since data enciphering and data encoding are implemented in two separate steps, this approach has the disadvantage of inefficiency in the implementation. Combining these two steps into one may obtain faster and more efficient implementation.

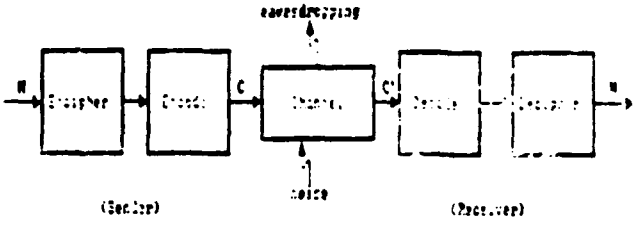


Fig 1 Conventional approach to data reliability and data secrecy

Joint encryption and error correction (JEEC) scheme

In his public-key cryptosystem, McEliece applied error-correcting capability of Goppa codes to provide data secrecy [8]. His idea is to introduce a random error vector to each encoded block before transmission. The Hamming weight (t') of the error vector is equal to the number (t) of errors the code can correct. Therefore, the receiver can remove the error vector and recover the plaintext by applying the decoding of the code. Any unauthorized user cannot do so without decoding keys because the general decoding problem for linear codes is NP-complete [9].

Evidently, if $t' < t$, then up to $t-t'$ errors may occur in the channel and these errors can be corrected by the receiver. Thus, the system can provide both data secrecy and data reliability simultaneously. Since there is a tradeoff between data secrecy and data reliability, large distance codes and large block length codes are required in this scheme [10,11]. Therefore, the scheme has the disadvantages of low information rate and high decryption overhead.

Definition 1. *The JEEC scheme*

A scheme that combines data encryption with data encoding into one enciphering process while providing a

This Paper was presented in CTYPTO'88 at University of California, Santa Barbara, Aug 21-25, 1988.

Paper No. 116-D;

This work was supported in part by the Office of Naval Research under Grant N00014-91-J-1067.

COMPUTER

Pseudorandom Bit Generators in Stream-Cipher Cryptography

Kencheng Zeng, Chung-Huang Yang, Dah-Yea Wei, and T.R.N. Rao
University of Southwestern Louisiana

The art of cryptography as a means for protecting private information against unauthorized access is as old as writing itself. Cryptography, indeed, is the only practical means for sending information over an insecure channel, be it telephone line, microwave, or satellite. The increasing use of electronic means of data communications, coupled with the growth of computer usage, has extended the need to protect information.

Stream ciphers play an especially important role in cryptographic practices — both diplomatic and military — that protect communications in the very high frequency domain. The central problem in stream-cipher cryptography, however, is the difficulty of generating a long unpredictable sequence of binary signals from a short and random key. Unpredictable sequences are desirable in cryptography because it is impossible, given a reasonable segment of its signals and computer resources, to find out more about them. Pseudorandom bit generators have been widely used to construct these sequences.

Considerable progress has been made in the design and analysis of pseudorandom bit generators over the last decade. The purpose of this article is to survey some of these developments.

The information age lends new dimensions to the art of cryptography. Techniques for encryption, decryption, and fending off attacks from intruders provide the only protection of sensitive data.

Background

To provide the general background for our exposition, we begin by describing a cipher (see Figure 1). A cipher conceals the *plaintext M* by transforming it into a disguised form, called the *ciphertext C*, so that only the authorized receiver can transform it back to the original plaintext. The process of transforming plaintext into ciphertext is called *encryption* or *enciphering*, and the inverse transformation from ciphertext to plaintext is called *decryption* or *deciphering*.

To prevent the plaintext from being easily revealed by an unauthorized person, the sender must transform a given plaintext into a large variety of possible ciphertexts selected by a specific parameter. This parameter is called the *encryption key* K_e. The receiver then deciphers the ciphertext using the *decryption key* K_d. In a public-key cryptosystem, K_e is made public while K_d is kept secret; it is computationally infeasible to deduce K_d from K_e. In a private-key cryptosystem, the sender and the receiver

0018-9162/91/0200-0008$01.00 © 1991 IEEE

This work was supported in part by the Office of Naval Research under Grant N00014-91-J-1067.

d-Functions in $V_k(F_2)$ and Self-decimation of m-Sequences

Kencheng Zeng[1], Dah-Yea Wei[2], and T.R.N. Rao[2]

Abstract. *With the purpose of generating, by the help of a single clock-controlled LFSR, a large class of binary sequences strong enough for cryptographic application, an extension class of key-specifiable transfer functions is proposed and analyzed. When tapped as feedforward networks to the LFSR under consideration, these functions will control the stepping of the latter in such a way that its output signals will be decimated pseudo-randomly at a pre-assigned rate. The decimation rate and the length of the LFSR are then suitably selected so that the resulting sequence will have, among others, a large prime period and a linear complexity comparable in order of magnitude to the period. Some conclusions concerning the average and maximal implementable decimation rates are also given.*

1. Introduction

On the basis of an idea put forth by R. Rueppel [1], W. G. Chambers and D. Gollmann [2] proposed a scheme, as shown in Figure 1, to decimate the signals of an m-sequence so as to produce, for suitably chosen parameters, a binary sequence with a prime period p and linear complexity equal to $p-1$ or p.

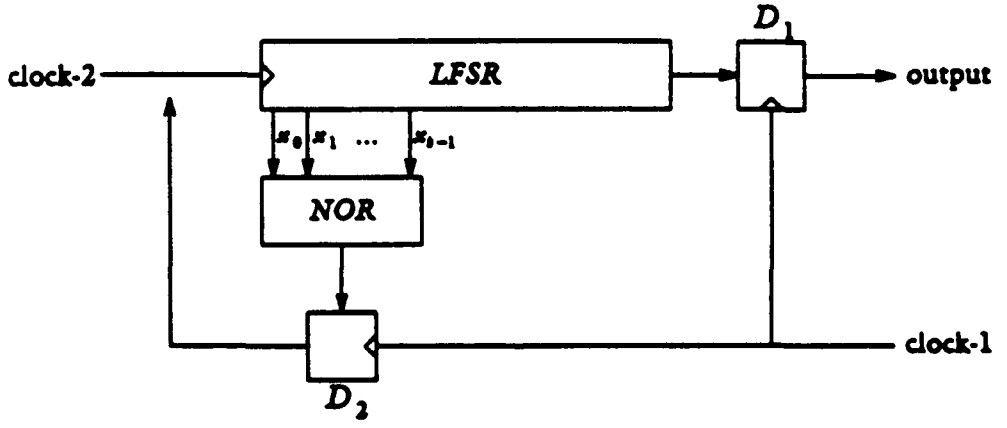


Figure 1

The scheme consists of an n-stage *LFSR* with an alterable primitive feedback polynomial and a k-place *NOR* function tapped to consecutive positions on it, together with two delay elements D_1, D_2 attached respectively to the output ends of the *LFSR* and the transferring *NOR* function. The delay elements are controlled by *clock-1* which works at a fixed speed determined by the channel of transmission,

[1] Graduate School of USTC, Academia Sinica, P.O. Box 3908, Beijing, People's Republic of China.
[2] The Center for Advanced Computer Studies, University of Southwestern Louisiana, P.O. Box 44330, Lafayette, Louisiana 70504-4330.

This work was supported in part by the Office of Naval Research under Grant N00014-91-J-1067.

An SFS Berger Check Prediction ALU and Its Application to Self-Checking Processor Designs*

Jien-Chung Lo
Department of Electrical Engineering
The University of Rhode Island
Kingston, RI 02881

Suchai Thanawastien †
Department of Computer Science
Rangsit University
Bangkok, Thailand

T. R. N. Rao †
The Center for Advanced Computer Studies
University of Southwestern Louisiana
Lafayette, LA 70504-4330

Michael Nicolaidis
TIM3/IMAG/INPG
Reliable Integrated Systems Group
46 Avenue Felix-Viallet
38031 Grenoble Cedex, France

Abstract

A strongly fault-secure (SFS) ALU design based on the Berger check prediction (BCP) technique is presented in this paper. The fault and error models of a large class of VLSI ALU designs are discussed. The proposed design is proved to be fault-secure and self-testing with respect to any single fault in the ALU part. Then, the proposed BCP ALU is proved to be SFS with any design of BCP circuit. Consequently, a self-checking processor whose data path is encoded entirely in a Berger code can be achieved. An efficient self-checking processor can then be designed.

* Accepted for publication "IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems".
† The research contributions of these authors are supported by ONR Grant #N00014-91-J-1067.

A DESIGN METHOD FOR TOTALLY SELF-CHECKING AND CODE-DISJOINT PLA OF COMBINATIONAL CIRCUITS*

T. R. N. Rao, G. L. Feng, and D. Y. Wei

The Center for Advanced Computer Studies

University of Southwestern Lousiana

Lafayette, LA. 70504

Abstract

Self-checking systems design is an important technique for concurrent error detection in fault-tolerant digital systems. The well-known concepts for self-checking systems are totally self-checking (TSC) circuits, which is defined to be both fault secure and self-testing, and code-disjoint (CD) circuits. The ideal self-testing systems should consist of several subsystems which are all TSC and CD. This paper presents a design method of TSC and CD PLA for combinational circuits. In this design, the input and output of a combinational circuit are encoded in systematic unordered codes. It should be mentioned that TSC checkers actually are all TSC and CD combinational circuits. Therefore, the TSC checkers using PLA can be obtained by this design method.

* This work was supported in part by the Office of Naval Research under Grant N00014-91-J-1067.

Decoding Algebraic–Geometric Codes up to the Designed Minimum Distance

Gui-Liang Feng and T. R. N. Rao, *Fellow, IEEE*

Abstract— **A simple decoding procedure for algebraic–geometric codes $C_\Omega(D, G)$ is presented. This decoding procedure is a generalization of Peterson's decoding procedure for the BCH codes. It can be used to correct any $\lfloor (d^* - 1)/2 \rfloor$ or fewer errors with complexity $O(n^3)$, where d^* is the designed minimum distance of the algebraic–geometric code and n is the codelength.**

Index Terms— **Error-correcting codes, algebraic–geometric codes, decoding procedure, correcting $\lfloor (d^* - 1)/2 \rfloor$ errors.**

I. Introduction

THE MOST important development in the theory of error-correcting codes in recent years is the introduction of methods from algebraic geometry to construct linear codes. These so called *algebraic–geometric codes* were introduced by Goppa. In 1982, Tsfasman, Vlăduţ and Zink [1] obtained an extremely exciting result: the existence of a sequence of codes that exceeds the Gilbert–Varshamov bound [2]. For this paper, they received the IEEE Information Theory Group Paper Award for 1983. Since then, many papers dealing with algebraic–geometric codes have followed [3]–[10].

Good code constructions are very important. Moreover, it is desirable and important to derive simple decoding procedures which can correct as many errors as possible. Justesen *et al.* [11] first presented a decoding procedure for codes from nonsingular plane algebraic curves. This decoding procedure can only correct $\lfloor (d^* - g - 1)/2 \rfloor$ or fewer errors, where d^* is the designed minimum distance of the code and g is the genus of the curve involved in the construction. Skorobogatov and Vlăduţ [12] generalized their ideas and gave a decoding procedure which can correct any $\lfloor (d^* - g - 1)/2 \rfloor$ or fewer errors for codes from arbitrary algebraic curves. In their paper, Skorobogatov and Vlăduţ also presented a modified algorithm, correcting more errors, but in general, not up to the designed minimum distance. Using profound results from algebraic geometry, Pellikaan [13] gave a decoding procedure which decodes up to $\lfloor (d^* - 1)/2 \rfloor$ errors. However, his decoding procedure is very complex and is not completely effective. Recently, Justesen *et al.* [14] improved on their original decoding procedure in several ways and gave a new decoding procedure for codes from arbitrary regular plane curves, which can decode up to $\lfloor (d^* - g/2 - 1)/2 \rfloor$ errors.

Manuscript received November 5, 1991; revised May 20, 1992. This work was supported in part by the Office of Naval Research under Grant N00014-91-J-1067. This work was presented in part at the 9th International Symposium for Applied Algebra, Algebraic Algorithms, and Error-Correcting Codes, New Orleans, LA, October 1991.

The authors are with the Center for Advanced Computer Studies, University of Southwestern Louisiana, Lafayette, LA 70504.

IEEE Log Number 9203437.

In this paper, we present a fairly simple decoding procedure capable of decoding up to $\lfloor (d^* - 1)/2 \rfloor$ errors. The improvement is obtained by using a form of majority scheme to find unknown syndromes in the well-known algorithm. The procedure can be implemented easily by hardware or software.

The paper is organized as follows. In the next section, for easy reference, we include a fundamental iterative algorithm (FIA), which is very similar to the Gaussian elimination and can be used to easily derive the Berlekamp–Massey algorithm and the generalized Berlekamp–Massey algorithm [16]. Then we modify the FIA and give some related properties, which will be used in other sections. In Section III, a new decoding procedure for algebraic–geometric codes $C_\Omega(D, G)$ with $G = mQ$ is presented. In order to easily understand this decoding procedure, one example is shown in Section IV. Finally, some conclusions are given in Section V.

II. Fundamental Iterative Algorithm

In this section, the fundamental iterative algorithm (FIA) [16] is modified. This modified algorithm is our main algorithm for decoding algebraic–geometric codes up to the designed minimum distance. To a certain extent it is similar to the Berlekamp–Massey algorithm, which is the main algorithm for decoding BCH codes up to the designed minimum distance. For easy reference, the FIA is described briefly in the following. This algorithm is for finding the smallest initial set of dependent columns in a matrix over any field F. That is, let

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1,N} \\ a_{21} & a_{22} & \cdots & a_{2N} \\ | & | & \cdots & | \\ a_{M1} & a_{M_2} & \cdots & a_{M,N} \end{bmatrix}$$

be such a matrix, to find the smallest l and $c_1, \cdots, c_l$ such that

$$a_{i,l+1} + c_1 \cdot a_{i,l} + \cdots + c_l \cdot a_{i,1} = 0 \qquad \text{for } i = 1, 2, \cdots, M. \tag{2.1}$$

For each column j, let $C^{(i-1,j)}(x) = \sum_{k=0}^{j-1} c_k^{(i-1,j)} x^k$, where $c_0^{(i-1,j)} = 1$, be defined as the polynomial with the property that

$$\left[C^{(i-1,j)}(x) \cdot a^{(h)}(x)\right]_j = a_{h,j} + c_1^{(i-1,j)} a_{h,j-1} + \cdots + c_{j-1}^{(i-1,j)} a_{h,1} = 0 \quad \text{for } h \leq i-1, \tag{2.2}$$

Error Correcting Codes over Z_{2^m} for Algorithm-Based Fault Tolerance *

G. L. Feng, T. R. N. Rao, and M. S. Kolluru

The Center for Advanced Computer Studies

University of Southwestern Lousiana

Lafayette, LA. 70504

Abstract

Algorithm-based fault tolerance is a scheme of low-cost error protection in real-time digital signal processing environments and other computation-intensive tasks. The basic idea of algorithm-based fault tolerance is encoding data at the system level and then redesigning the algorithm such that it can detect or correct errors in computation. In this paper, a new method for encoding data is proposed and furthermore, two kinds of error-correcting codes over Z_{2^m}, which can be used with fixed-point arithmetic in practical algorithm-based fault tolerant systems, are introduced. In addition, two simple decoding procedures are also proposed.

* This work was supported in part by the Office of Naval Research under Grant N00014-91-J-1067.

Novel Totally Self-Checking Berger Code Checker Designs Based on Generalized Berger Code Partitioning*

T. R. N. Rao, G. L. Feng, Mahadev S. Kolluru

The Center for Advanced Computer Studies
University of Southwestern Louisiana
Lafayette, LA. 70504

Jien-Chung Lo

Department of Electrical Engineering
University of Rhode Island
Kingston, RI. 02881

Abstract

Novel totally self-checking(TSC) Berger code checker designs are presented in this correspondence. We derive the generalized Berger check partitioning and prove that a TSC Berger code checker can be constructed from a TSC m-out-of-n checker. For a TSC Berger code checker design, no two-output checker exists for information length 2^{r-1}, for any positive non-zero r. The presented approach solves this open problem.

**This work was supported in part by the Office of Naval Research under Grant N00014-91-J-1067.*

Efficient Multiple Unidirectional Byte Error-Detecting Codes for Computer Memory Systems *

T. R. N. Rao, G. L. Feng and M. S. Kolluru
The Center for Advanced Computer Studies
University of Southwestern Louisiana
Lafayette, LA 70504-4330

Abstract

In this paper, a new method of construction of more efficient codes, which can detect t unidirectional byte errors or all unidirectional bit errors (t-UbED/AUED), is presented. In this construction, we generalize and improve the t-UbED/AUED codes proposed by Dunning et al., in such a way that two weight syndromes need not be protected from unidirectional errors when t > 2. Thus, this construction is more efficient and can be applied to all multiple unidirectional byte error-detecting codes.

Keywords: bit unidirectional errors, unidirectional byte errors, memory fault tolerance, error-detection codes.

I. Introduction

In computer memory systems, for data stored in a byte-per-chip or byte-per-card fashion, byte errors tend to occur [1-2]. Codes have been designed for byte-error correction and detection [3-7], and also for their detection and correction together with random errors [8-12].

The faults that occur in many computer memories and VLSI circuits, most likely cause "*unidirectional errors*", for which both 1→0 and 0→1 errors may occur, though not occurring simultaneously in a single data word. However, the errors in one byte may be of the form 1→0 while in another byte they may be the from 0→1. When a memory is handled as a whole, unidirectional errors of a single form may occur across the entire word. Figure 1.1 contrasts a unidirectional error with a unidirectional byte error affecting two bytes.

In byte-organized computer memories, transient and permanent faults are apt to cause multiple unidirectional byte errors. Codes that can detect a small number t of unidirectional byte errors in computer memory words, composed of small m information bytes each containing b bits, and which also detect simultaneously, all unidirectional errors across the entire memory word, were developed in [5-6]. These codes are termed t-unidirectional byte error detecting and all unidirectional error detecting (t-UbED/AUED) codes, and always use two additional b-bit bytes to hold the parity check information. Figure 1.2 shows the general form of a codeword including both information and check bytes.

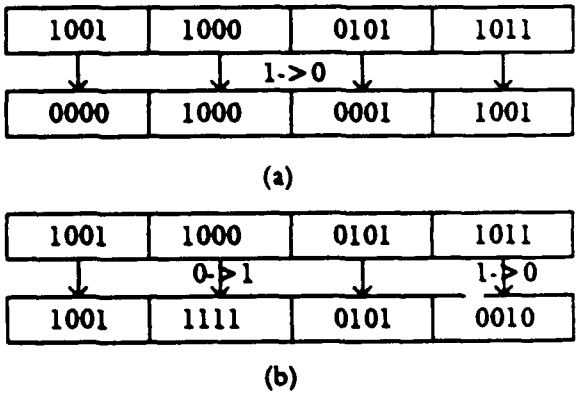


Fig. 1.1 (a) Unidirectional error example.
(b) 2-Unidirectional byte error example.

In this paper, we generalize the construction of t-UbED/AUED codes in [6] to any t and further improvise the construction for t ≥ 3. The organization of this paper is follows. In the next section, we briefly review the general principle of construction of t-UbED/AUED codes. Then we generalize the construction proposed by

* This work was supported in part by the Office of Naval Research under Grant N00014-91-J-1067.

0731-3071/92 $03.00 © 1992 IEEE

True Designed-Distance Decoding of a Class of Algebraic-Geometric Codes, Part I: Proving the Designed Distance without Riemann-Roch Theorem

G. L. Feng, V. K. Wei, T. R. Rao, and K. K. Tzeng

Abstract

A new decoding procedure for algebraic-geometric codes is presented. For codes from a large class of irreducible plane curves, including the Hermitian curve, it can correct up to $\lfloor \frac{d^* - 1}{2} \rfloor$ errors, where d^* is the designed minimum distance. With it we also obtain a proof of $d_{min} \geq d^*$ without using Riemann-Roch Theorem. In this part, we present the theory and the basic algorithm. In Part II, we present a fast implementation whose complexity is $O(n^{7/3})$.

Index Terms: algebraic geometry codes, decoding, Hermitian curve, Riemann-Roch Theorem, Gauss elimination.

G. L. Feng and T. R. Rao are with the Center for Advanced Computer Studies, University of Southwestern Louisiana, Lafayette, LA 70504. This work was supported in part by the Office of Naval Research under Grant N00014-91-J-1067.

V. K. Wei is with Bellcore, 445 South Street, Morristown, NJ 07960.

K. K. Tzeng is with the Department of Electrical Engineering and Computer Science, Lehigh University, Bethlehem, PA 18015. This work was support by the National Science Foundation under Grant NCR-9016095.

True Designed-Distance Decoding of a Class of Algebraic-Geometric Codes, Part II: Fast Algorithms and Block Hankel Matrices

*G. L. Feng**, *V. K. Wei*†, *T. R. Rao**, *and K. K. Tzeng*‡

Abstract

We present fast Gauss elimination algorithms for Hankel matrices and block Hankel matrices. In combination with other procedures, we obtain a true designed-minimum-distance decoding algorithm with complexity $O(n^{7/3})$ for the class of algebraic-geometric codes studied in Part I. These Gauss elimination algorithms are also useful in several other decoding and shift-register synthesis applications.

Index Terms: Algebraic geometry codes, decoding, Gauss elimination, Hankel matrix, block Hankel matrix, Shift-register synthesis.

*The first and the third authors are with the Center for Advanced Computer Studies, University of Southwestern Louisiana, Lafayette, LA 70504. This work was supported in part by the Office of Naval Research under Grant N00014-91-J-1067

†The second author is with Bellcore, 445 South Street, Morristown, NJ 07960.

‡The fourth author is with the Department of Electrical Engineering and Computer Science, Lehigh University, Bethlehem, PA 18015 This work was support by the National Science Foundation under Grant NCR-9016095.

The Efficient Design of a Strongly Fault-Secure ALU Using a Reduced Berger Code for WSI Processor Arrays†

J. H. Kim, T. R. N. Rao, and G. L. Feng
The Center for Advanced Computer Studies
University of Southwestern Louisiana
Lafayette, LA 70504-4330

J.-C. Lo
Dept. of Electrical Eng.
University of Rhode Island
Kingston, RI 02881

Abstract

A self-checking processor (SCP) is a processor that is designed with concurrent error detection (CED) capability. CED is an error/fault detection process that is designed to operate concurrently with the normal processor operations. CED is a very important and necessary feature in WSI processor arrays that are integral to real-time applications. Due to its operative nature, arithmetic and logic units (ALUs) are the most difficult functional circuit to check among the components of a processor. In this paper, we present an efficient design of a 32-bit strongly fault-secure (SFS) ALU using a Reduced Berger code. A reduced Berger code is used to encode both operands and the computation results. This reduced Berger code uses only the two least significant check bits of its Berger code counterpart regardless of information length. Since a Berger code requires $\lceil \log_2(n+1) \rceil$ check bits for n information bits, the application of reduced Berger code yields more efficient implementation of a strongly fault-secure ALU than the previously proposed techniques.

1. Introduction

The complexity of an IC chip increases significantly as a result of the advent of very large scale integrated (VLSI) technology. A modern microprocessor built on a single VLSI chip is more complex than a medium scale computer just a few years ago. Since the future WSI circuits should be more dense with smaller feature sizes, the permanent and transient faults are more likely to occur in the future WSI circuits than that at the present time. Concurrent error detection (CED) is thus vital for the success of future WSI development.

A self-checking processor (SCP) is a processor that is designed with concurrent error detection (CED) capability. CED is an error/fault detection process that is designed to operate concurrently with the normal processor operations. CED is a very important and necessary feature in WSI processor arrays that are integral to real-time applications, since the error latency time will be very small so as to enable fast error recovery and to prevent system crashes. SCP can be very effective in fault-tolerant computer system design. Important classes of SCP include: a totally self-checking (TSC) processor and a strongly fault-secure (SFS) processor. A typical TSC or SFS processor consists of a TSC or SFS

†This research is partly supported by NSF grant NSF-ADP-04 and by Board of Regents of Louisiana grant LEQSF RD-A-26

This work was supported in part by the Office of Naval Research under Grant N00014-91-J-1067.

A Simple Approach for Construction of Algebraic Geometric Codes from Affine Plane Curves

G. L. Feng and T. R. N. Rao* Fellow IEEE*

Abstract

The current algebraic-geometric (AG) codes are based on the theory of algebraic geometric curves. In this paper, we present a simple approach for the construction of AG codes which does not require an extensive background in algebraic geometry. Given an affine plane irreducible curve and all its rational points, we can find a sequence of monomials $x^i y^j$ based on the equation of the curve. Using the first r monomials as a basis for the dual code of a linear code, the designed minimum distance d of the linear code, called AG code, can be easily determined. For these codes, we show a fast decoding procedure with a complexity $O(n^{7/3})$, which can correct errors up to $\lfloor (d-1)/2 \rfloor$. For this approach it is neither necessary to know the genus of curve nor the basis of a differential form. This approach can be easily understood by most engineers.

Index Terms: algebraic geometric codes, fast decoding, minimum distance, error-correcting codes.

* G. L. Feng and T. R. N. Rao are with the Center for Advanced Computer Studies, University of Southwestern Louisiana, Lafayette, LA. 70504, USA. This work was supported in part by the Office of Naval Research under Grant N00014-91-J-1067.

A Class of Algebraic Geometric Codes from Curves in High-Dimensional Projective Spaces

G. L. Feng and T. R. N. Rao Fellow IEEE
the Center for Advanced Computer Studies,
University of Southwestern Louisiana, Lafayette, LA. 70504, USA

Abstract

Most of the research work in the area of algebraic geometric (AG) codes deals with the construction of AG codes from plane algebraic geometric curves. But, some work pertains to the construction of the AG codes from non-planar algebraic geometric curves. However, longer AG codes must have relatively larger genus and should only be the codes constructed from non-planar curves. In this paper, we present a new construction of a class of AG codes from curves in high-dimensional projective spaces. For this construction, it is easy to determine the designed minimum distance and find the parity check matrix, and the decoding up to the designed minimum distance is fast. Furthermore, this approach can be easily understood by most engineers.

I. Introduction

The most important development in the theory of error-correcting codes in recent years is the introduction of methods from algebraic geometry to construct linear codes [1-3]. These so-called *algebraic geometric codes (AG codes)* were introduced by Goppa in 1980. In 1982, Tsfasman, Vladut and Zink [4] obtained an extremely exciting result: the existence of a sequence of AG codes which exceeds the Gilbert-Varshamov bound [5]. For this paper they received the IEEE Information Theory Group Paper Award for 1983. Since then, many papers dealing with algebraic geometric codes have followed [6-16]. However, most of these papers deal only with the AG codes obtained from plane curves.

The greatest advantage of AG codes is that they offer more flexibility in the choice of code parameters. Most importantly, for a fixed finite field $\mathbf{F}_q$, there are AG codes having any length. It is known that the coordinates of AG codes are the rational points of algebraic geometric curves. Thus, greater the number of rational points, longer the length of the AG code. Over $\mathbf{F}_q$, the number of rational points of any plane algebraic geometric curve is obviously less than $q^2 + q + 1$. Thus, actually useful AG codes should be the AG codes constructed from curves in high-dimensional projective spaces (HDAG codes). In [17], Pellikaan et. al constructed a large class of codes from non-planar curves. Justesen et. al in [6], first gave a description of algebraic-geometric codes defined only by monomials. Following this description of AG codes, in [18], a simple approach for the construction of AG codes from affine plane algebraic geometric curves has been proposed. This new approach can be easily understood by most engineers. In this paper, we generalize the results in [18] to the case of curves in high-dimensional projective spaces and further present a construction of a class of HDAG codes. The codes considered here are essentially the algebraic-geometric Reed-Muller codes [19].

This work was supported in part by the Office of Naval Research under Grant N00014-91-J-1067.

ERASURES-AND-ERRORS DECODING OF ALGEBRAIC-GEOMETRIC CODES

G. L. Feng and T. R. N. Rao
The Center for Advanced Computer Studies
University of Southwestern Louisiana
Lafayette, LA. 70504

Abstract

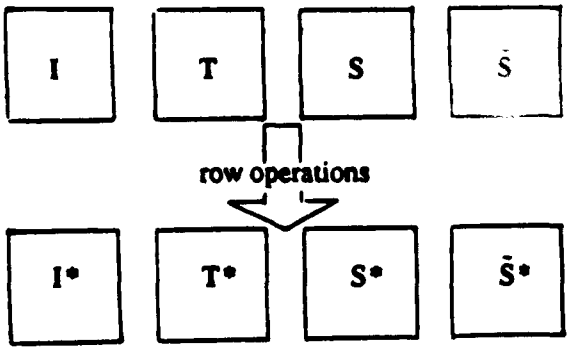
It is well known that the receiver with an erasure option can improve the probability of decoding error. It is desirable to find as simple a decoding algorithm as possible for correcting erasures as well as errors. Such an algorithm enables one to carry out a generalized minimum distance decoding for further improving the probability of decoding error. Forney [1] first formulated an erasures-and-errors decoding algorithm for BCH codes. Berlekamp [2] formulated an elegant erasures-and-errors decoding algorithm for BCH codes based on his error-only decoding algorithm. For Goppa codes, Sugiyama et. al presented an erasure-and-error decoding algorithm in [3]. In this paper, we present an erasures-and-error decoding procedure for AG codes, that includes the errors-only decoding procedure [4-6] as a special case.

Let the number of erasure locations be ρ and let the erasure locations be $P_{\pi(1)}, P_{\pi(2)}, \ldots, P_{\pi(\rho)}$, where $\rho < d^*$ and d^* is the designed minimum distance. Thus, any pattern of ν errors and ρ erasures can be decoded, provided $2\nu + 1 + \rho \le d^*$ is satisfied.

For convenience, let us consider the AG codes from plane curves with a basis of monomials as shown in [5-7]. In this case, $P_{\pi(\mu)} = (x_{\pi(\mu)}, y_{\pi(\mu)})$. For general AG codes, it is straightforward. Let $T_{i,j} = \sum_{\mu=1}^{\rho} x^i_{\pi(\mu)} y^j_{\pi(\mu)}$, and let T be this matrix, in which the first row and the first column consist of elements of $\{ T_{i,j} \mid \underline{x^i y^j}$ are in increasing order $\}$, i.e., $\{ x^i y^j \}$ are the first $d^* + g - 1$ monomials of $H^{(1)}$ or $H^{(2)}$, and in which if the entry at the first row and column k of T is supposed to be $T_{u,v}$ and the entry at the first column and row h of T is supposed to be $T_{i,j}$, then the entry at row h and column k of T is $T_{i+u,j+v}$.

Let S be a syndrome matrix and $\tilde{S}$ be a matrix modified from S, by substituting zero for all unknown syndromes. Let I be the $(d^* + g - 1) \times (d^* + g - 1)$ identity matrix.

Appling the Gauss elimination to matrix T by only row operations, we obtain T* from T. Performing the same row operations, I is transformed to I*, S to S*, and $\tilde{S}$ to $\tilde{S}^*$. Then each row in I* corresponding to the zero row of T*, is an erasure locator polynomial. The matrix S** obtained by deleting the rows in S*, which correspond to the nonzero rows in T*, is called a modified syndrome matrix for the erasure locations. $\tilde{S}^*$ is called a discrepancy matrix.



This work was supported in part by the Office of Naval Research under Grant N00014-91-J-1067.